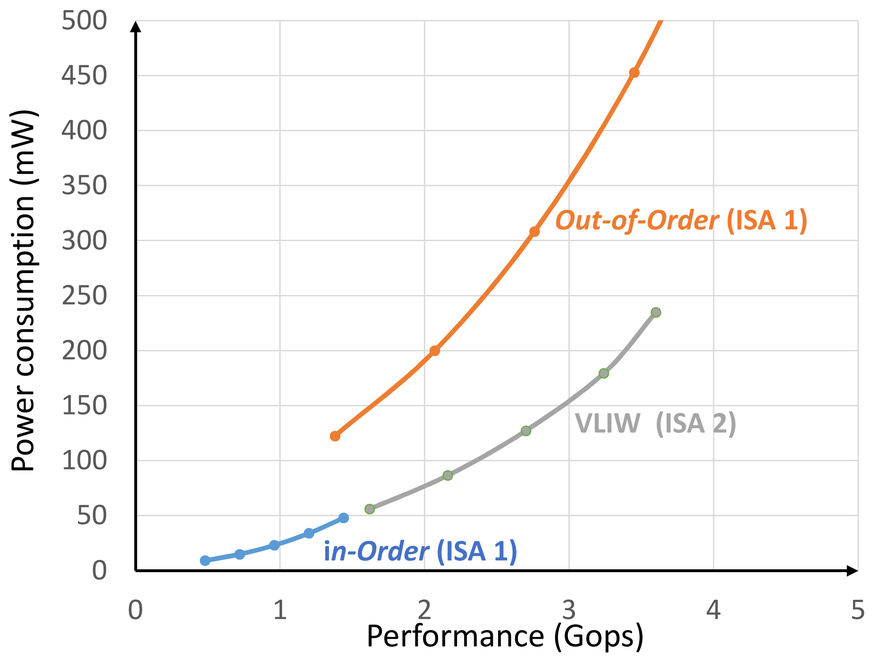
Hybrid-DBT: Hardware Accelerated Dynamic Binary Translation

Simon Rokicki, Erven Rohou, Steven Derrien

Hybrid-DBT: Hardware Accelerated Dynamic Binary Translation

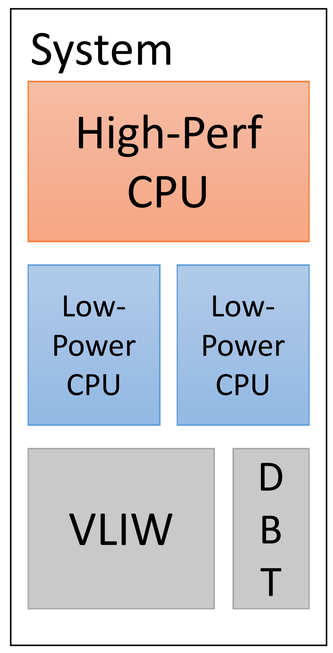
Simon Rokicki, Erven Rohou, Steven Derrien
Univ Rennes, Inria, CNRS, IRISA

Context - Heterogeneous Multi-cores



Heterogeneous multi-core systems present key advantages compared to their homogeneous counterparts. They allow a dynamic balancing between performance and energy efficiency. Current implementations are limited to a single ISA to easily migrate tasks from one core to the other.

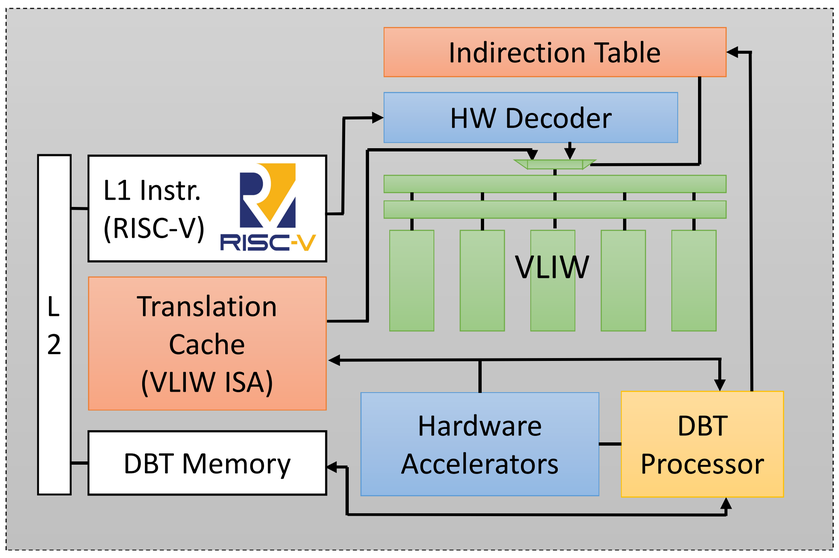
To further improve the current systems, specialized cores such as VLIWs can be added. To handle the ISA differences between the cores, a layer of Dynamic Binary Translation is associated to the specialized cores. DBT translates the instructions from a given ISA to another one as they are being executed on the target core. To lessen the overheads introduced by the use of DBT, we present Hybrid-DBT: a HW/SW Co-Designed DBT system which uses several hardware accelerators to reduce the costs of DBT [3][4].



HW/SW Co-Designed Machines

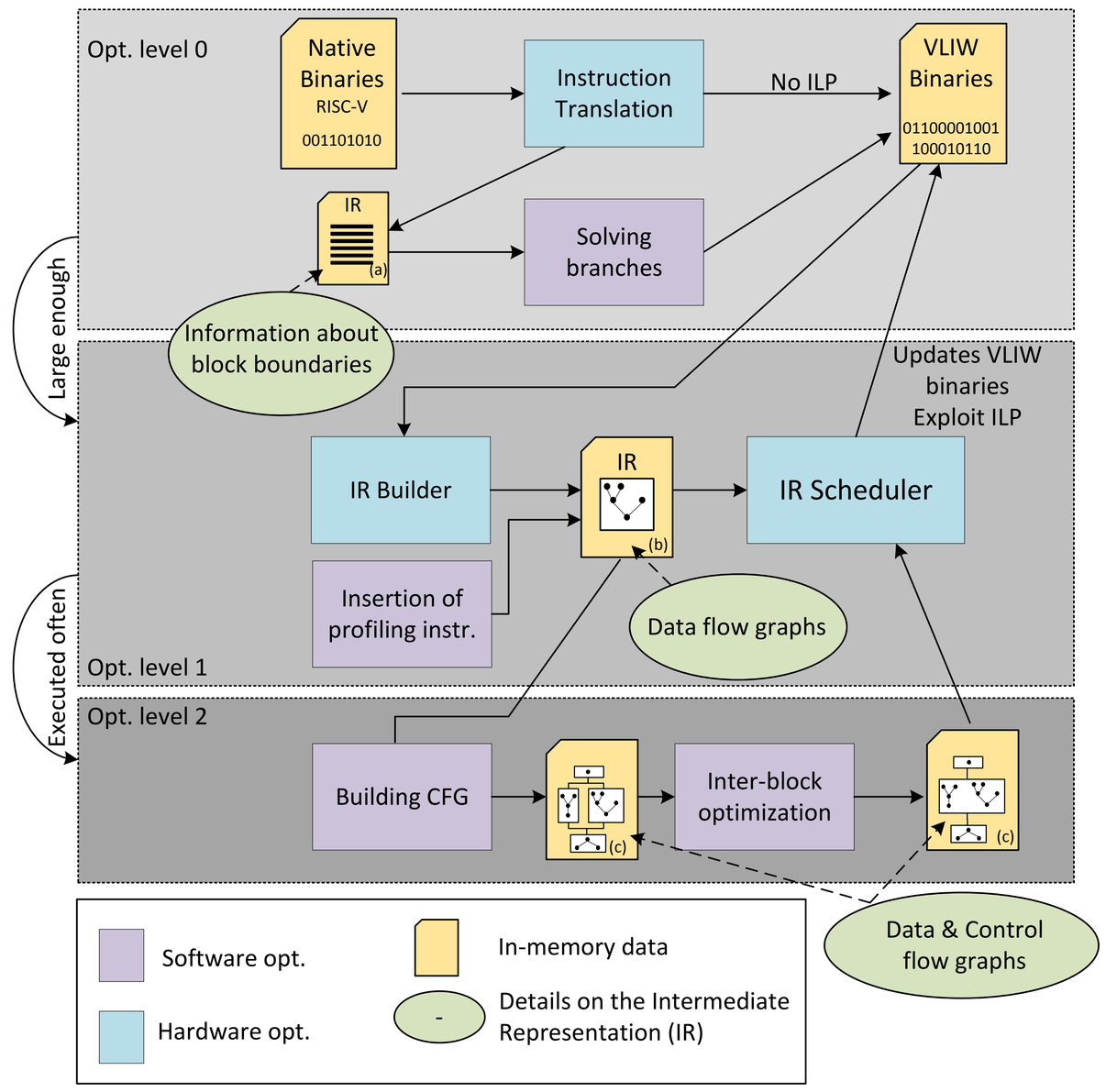
Previous work on HW/SW Co-Designed Machines:
→ Transmeta Crusoe (x86 on VLIW) [1]
→ NVidia Denver (Armv8 on VLIW) [2]

Overview of Hybrid-DBT

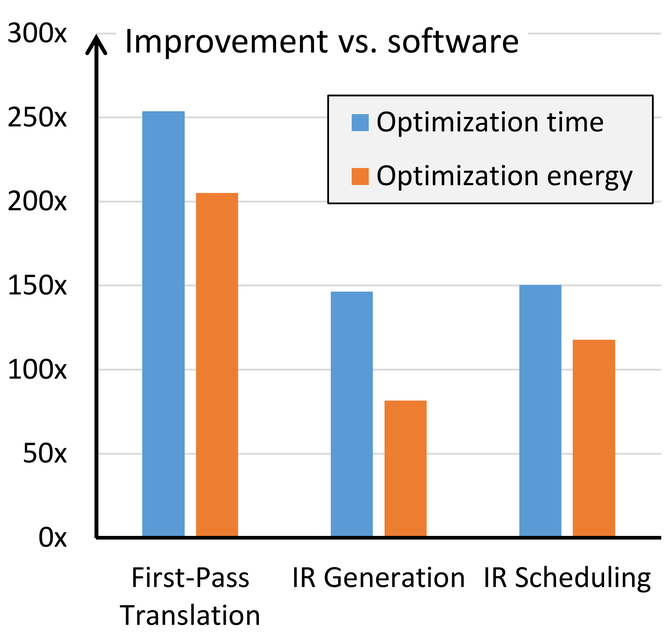


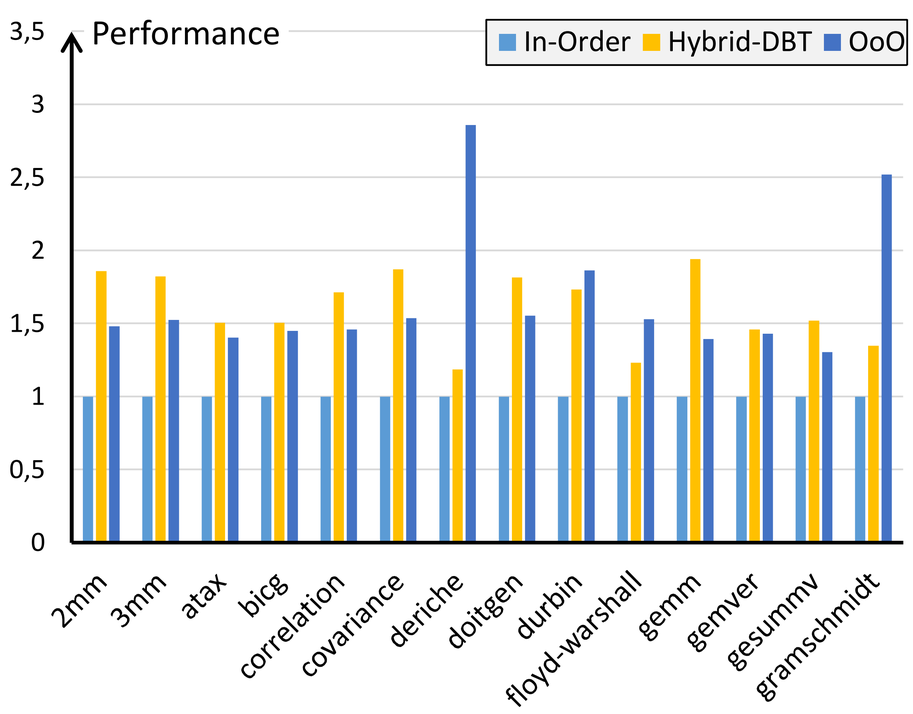
→ The **VLIW core** executes applications
→ The **L1 Instr.** cache stores RISC-V instructions
→ The **HW Decoder** decodes RISC-V into VLIW ISA
→ The **DBT Processor** analyses and optimizes RISC-V binaries, using the **HW Accelerators**.
→ Those binaries are stored in the **Translation Cache**.

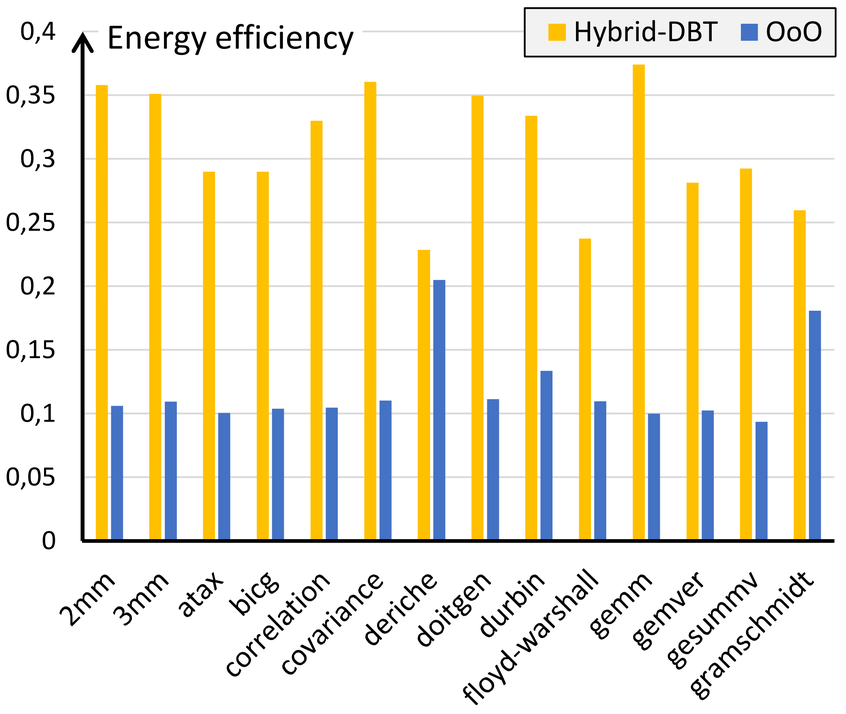
Translation flow



Experimental Results







References

[1] Dehnert et al. *The Transmeta Code Morphing Software: Using Speculation, Recovery, and Adaptive Retranslation (...)*
[2] Boogs et al. *Denver: NVidia's First 64-bit ARM Processor*
[3] Rokicki et al. *Hybrid-DBT: HW/SW DBT targeting VLIW*
[4] Rokicki et al. *Supporting Runtime Reconfigurable VLIWs Cores through Dynamic Binary Translation*

Contacts

simon.rokicki@irisa.fr
https://github.com/srokicki/HybridDBT